CEWES Major Shared Resource Center

MSRC

CEWES MSRC/PET TR/99-04

UTPROJ: The University of Texas Projection Code for Computing Locally Conservative Velocity Fields

by

Mary F. Wheeler
Clint Dawson
Jichun Li
Victor Parr

UTPROJ: The University of Texas Projection Code for Computing Locally Conservative Velocity Fields

Mary F. Wheeler, Clint Dawson, Jichun Li and Victor Parr*

March 30, 1999

1 Introduction

In the numerical modeling of fluid flow and transport problems, the computed velocity field frequently needs to be projected from one grid to another between different models. Usually the flow and multi-species transport are solved separately using completely different numerical methods and grids due to differences in length and time scales of the phenomena involved. For accurate transport, it is often desirable for the velocities to be locally conservative on the transport grid. If we do not have local mass conservation it amounts to adding spurious sources and sinks to the transport equation [4]. Local mass conservation can be accomplished through a projection algorithm.

One particular example where projection is needed is in the modeling of surface flow. The velocities computed from a flow model, e.g., ADCIRC [8] (an advanced circulation model for shelves, coasts and estuaries based on unstructured triangular grids) or RMA [7] (also uses unstructured triangular grids), are used as input to a transport model, such as CE-QUAL-ICM [3] (a water quality model using unstructured quadrilateral grids). These codes are widely used by U.S. Army Corps of Engineers at the Waterways Experiment Station in Vicksburg, Mississippi, and other state and federal agencies in environmental quality modeling. Therefore, there is a need to couple these flow and water quality models and perform a projection to produce a locally conservative velocity field on the transport grid.

This paper describes UTPROJ, the University of Texas PROJection code, which constructs locally mass conservative velocity fields by solving scalar linear second order elliptic equations. The code is built on TUF (the Texas Unstructured Flow program in C++) [5] and an algorithm described in [4]. TUF was developed to solve elliptic equations on general unstructured meshes in two and three space dimensions using mixed finite element methods [2]. The user interface to UTPROJ is based on Keenan's Kscript package [6], a powerful and plain English like script laguage. The input to UTPROJ3D-2D is easily obtained through a preprocessing of the output from the flow model in *Tecplot* (a widely used graphics package) format. In the next sections, we describe the numerical formulation used in UTPROJ and give some numerical examples.

*Texas Institute for Computational and Applied Mathematics, The University of Texas at Austin, 105 Dean Keaton Street; C0200, Austin, Texas 78712.

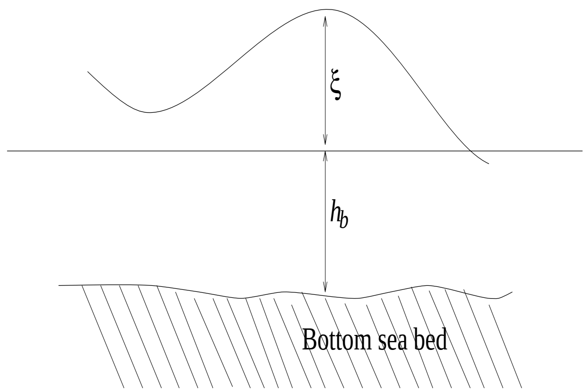

Figure 1: Definitions of ξ, h_b and h

2 Details of the Numerical Algorithm

To be consistent, we use the same notations as in [4]. Let $\Omega \in R^n$, n = 2 or 3, be the physical domain which is being modeled and $\partial\Omega$ the external boundary of this domain.

The general form of the conservation of mass in modeling surface water flow is given by[10]:

$$\frac{\partial h}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot_{xy}(h\mathbf{U}) = \mathrm{q}, \quad \text{for 2D}, \tag{1}$$

or

$$\boldsymbol{\nabla}\cdot\mathbf{U} = \mathrm{q}, \quad \text{for 3D}. \tag{2}$$

In the above equation, $h = \xi + h_b$ is the surface elevation (see Figure 1), $\mathbf{U}$ is the velocity vector field in 3D (or the depth averaged velocity field in 2D), $\boldsymbol{\nabla}$ (or $\boldsymbol{\nabla}_{xy}$) is the spatial gradient operator in 3D (or 2D) and q represents the sources and sinks that may be present in the flow domain.

Let h^o be the mesh parameter of the old grid and h^n be the mesh parameter of the new grid, where the projected velocities are desired. Further, let $\mathbf{V}_{h^o}$ and $\mathbf{V}_{h^n}$ be the finite dimensional subspaces corresponding to the old and new meshes. The main idea of our projection algorithm is to find a locally mass conservative velocity $\mathbf{U}_{h^n} \in \mathbf{V}_{h^n}$ from $\mathbf{U}_{h^o} \in \mathbf{V}_{h^o}$. For this purpose, the new velocity $\mathbf{U}_{h^n}$ is expressed in terms of the old velocity $\mathbf{U}_{h^o}$ in the following manner:

$$\mathbf{U}_{h^n} = \mathcal{P}_{h^n}\mathbf{U}_{h^o} + \boldsymbol{\Gamma}_{h^n} \quad \in \mathbf{V}_{h^n}, \tag{3}$$

where $\mathcal{P}_{h^n}\mathbf{U}_{h^o}$ is the $\mathcal{L}^2$ (or other equivalent) projection into $\mathbf{V}_{h^n}$ of the old velocity $\mathbf{U}_{h^o}$ and , $_{h^n} \in \mathbf{V}_{h^n}$ is the velocity correction which we need to compute.

Generally speaking, we want $\mathbf{U}_{h^n}$ to satisfy the mass conservation law given by:

$$\begin{aligned} \boldsymbol{\nabla}\cdot\mathbf{U}_{h^n} &= f \quad \text{in } \Omega, \\ \mathbf{U}_{h^n}\cdot\boldsymbol{\nu} &= \mathrm{g} \quad \text{on } \partial\Omega. \end{aligned} \tag{4}$$

Substituting (3) into (4), we obtain the following boundary value problem:

$$\begin{aligned} &\boldsymbol{\nabla}\cdot\boldsymbol{\Gamma}_{h^n} = \mathrm{f} - \boldsymbol{\nabla}\cdot\mathcal{P}_{h^n}\mathbf{U}_{h^o} = \tilde{\mathrm{f}} \quad \text{in } \Omega, \\ &\boldsymbol{\Gamma}_{h^n}\cdot\boldsymbol{\nu} = 0 \ \text{ on } \partial\Omega. \end{aligned} \tag{5}$$

To solve the problem, we denote $\mathbf{\Gamma}_{h^n} = -\mathbf{\nabla}\phi_{h^n}$, where the scalar variable ϕ_{h^n} is a pseudo-pressure. Hence we obtain the following elliptic problem:

$$\begin{aligned} -\triangle\phi_{h^n} &= \tilde{\mathrm{f}} \qquad \text{on} \;\; \Omega, \\ -\mathbf{\nabla}\phi_h\cdot\boldsymbol{\nu} &= 0 \qquad \text{on} \;\; \partial\Omega. \end{aligned} \tag{6}$$

The elliptic problem given by (6) is solved by using the mixed/hybrid finite element method which approximates both fluxes $\mathbf{\Gamma}$ and pressures ϕ. In addition, the fluxes $\mathbf{\Gamma}\cdot\boldsymbol{\nu} = -\mathbf{\nabla}\phi\cdot\boldsymbol{\nu}$ are continuous across the edges and the resulting numerical solution satisfies mass conservation cell-by-cell. The mixed/hybrid finite element approximation of the elliptic problem (6) along with velocity relations (3) represent the *conservative velocity projection formulation.*

On the new grid, the elliptic problem is solved using the lowest order Raviart-Thomas spaces:

$$\mathrm{W}_{h^n}(E) = \{a \in \mathrm{R} \;\; \text{on} \;\; E\}, \tag{7}$$

$$\mathbf{V}_{h^n}(E) = \{(\alpha + \beta \mathrm{x}, \gamma + \beta y, C_1 + C_2 z)^T; \alpha, \beta, \gamma, C_1, C_2 \in \mathrm{R}, \quad (\mathrm{x}, \mathrm{y}, z) \in E, \tag{8}$$

and

$$\Lambda_{h^n}(\partial E) = \{a \in \mathrm{R} \;\; \text{on} \;\; \partial E\}, \tag{9}$$

where E is any given (triangular, rectangular, tetrahedral, hexahedral or prismatic) element. Here $C_1 = C_2 = 0$ in 2D.

The finite dimensional scalar and vector spaces on the new grid are defined as:

$$\mathrm{W}_{h^n} = \left\{ w \in \mathcal{L}^2(\Omega) : w|_E \in \mathrm{W}_{h^n}(E), \quad \forall E \right\}, \tag{10}$$

$$\mathbf{V}_{h^n} = \left\{ \mathbf{v} \in (\mathcal{L}^2(\Omega))^d : \mathbf{v}|_E \in \mathbf{V}_{h^n}(E), \quad \forall E \right\} \tag{11}$$

and

$$\mathbf{\Lambda_{h^n}} = \{\mu : \mu|_{\partial\mathbf{E}} \in \mathbf{\Lambda_{h^n}}(\partial\mathbf{E}), \quad \forall\mathbf{E}\}. \tag{12}$$

Here d is the spatial dimension.

The mixed/hybrid finite element method for the second-order elliptic problem is:
Find $(\mathbf{\Gamma}_{h^n}, \phi_{h^n}, \lambda_{h^n}) \in (\mathbf{V}_{h^n}, W_{h^n}, \Lambda_{h^n})$ such that

$$\begin{array}{ll} (\mathbf{\Gamma}_{h^n}, \mathbf{v}_{h^n})_E - (\phi_{h^n}, \mathbf{\nabla}\cdot\mathbf{v}_{h^n})_E + \langle\lambda_{h^n}, \mathbf{v}_{h^n}\cdot\boldsymbol{\nu}_E\rangle_{\partial E} = 0 & \forall \;\; \mathbf{v}_{h^n} \in \mathbf{V}_{h^n}, \;\; \mathbf{v}_{h^n}\cdot\boldsymbol{\nu} = 0 \text{ on } \partial\Omega, \\ (\mathbf{\nabla}\cdot\mathbf{\Gamma}_{h^n}, w_{h^n})_E = \left(\tilde{\mathrm{f}}, w_{h^n}\right)_E & \forall \;\; w_{h^n} \in W_{h^n}, \\ \sum_{\partial E}\langle\mathbf{\Gamma}_{h^n}\cdot\boldsymbol{\nu}_E, \mu_{h^n}\rangle_{\partial E} = 0 & \forall \;\; \mu_{h^n} \in \Lambda_{h^n}, \\ \mathbf{\Gamma}_{h^n}\cdot\boldsymbol{\nu} = 0 \text{ on } \partial\Omega, & \end{array} \tag{13}$$

where $(\cdot,\cdot)_E$ and $\langle\cdot,\cdot\rangle_{\partial E}$ are the usual inner product on E and ∂E, and $\boldsymbol{\nu}_E$ is the unit outward normal to ∂E. In the third equation the sum is over all element boundaries. Note that this equation enforces the continuity of the normal flux across element edges in the interior of Ω.

Equations (13) gives rise to a linear algebraic system of the form

$$\begin{bmatrix} A & -B & C \\ B^T & 0 & 0 \\ C^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{\Gamma} \\ \mathbf{\Phi} \\ \mathbf{\Lambda} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{F} \\ 0 \end{bmatrix}, \tag{14}$$

where A is symmetric and positive definite, and B and C are rectangular. Eliminating $\mathbf{\Gamma}$ and $\mathbf{\Phi}$ in terms of $\mathbf{\Lambda}$, we obtain a symmetric positive definite system in $\mathbf{\Lambda}$ variables only. UTPROJ solves this system using diagonally preconditioned conjugate gradient iteration. Once $\mathbf{\Lambda}$ is known, then $\mathbf{\Phi}$ and $\mathbf{\Lambda}$ can be determined.

More information on the mixed/hybrid finite element method and implementation details can be found in [2].

3 Numerical examples

In this section, we give three numerical examples, two two-dimensional examples and one three-dimensional example.

The first problem is a standard test problem in hydrodynamics, the quarter annular harbor problem. The domain and finite element mesh for this example are given in Figure 2. On the inner circle of the annulus, a land boundary is prescribed and on the outer circle elevation is specified. The lateral boundaries were modeled as a solid wall. Velocities were generated using the ADCIRC hydrodynamics code. In this code, velocities and elevation are computed at the nodes of the finite element mesh. The velocities at a specified time are plotted at the nodes in Figure 2. For this velocity field, the local mass error was computed and plotted in Figure 3. As shown in this figure, the mass error is on the order of 10^3 to 10^4 for most of the elements. The local mass error after postprocessing is given in Figure 4, indicating that UTPROJ has reduced the error to about 10^{-5} over the entire domain. For this example, the code took 103 conjugate gradient iterations and 260 milliseconds to compute the solution.

The second problem is representative of coupling between a flow and transport algorithm. Here we model a scenario of a contaminant (e.g. oil) spill in the Houston Ship Channel. The physical domain and finite element mesh are given in Figure 5. A constant source of contaminant is released near the entrance to the ship channel. The contaminant concentration c, is assumed to satisfy the transport equation

$$\frac{\partial(hc)}{\partial t} + \mathbf{\nabla}\cdot_{xy}(h\mathbf{U}c) = 0. \tag{15}$$

Velocities were computed using the ADCIRC code simulating a period of two days using thirty second time steps. The transport solution was computed on the same grid, using 7.5 minute time steps. The velocities are time averaged before being projected. Transport is approximated using a one point upwinding scheme on each triangular element. This approach is similar to the finite volume method used in CE-QUAL-ICM, restricted to two space dimensions. A plot of the tracer solution after several hours of flow, is given in Figure 6, indicating the spread of contaminant through the coastal environment.

The final example is a three-dimensional example, with mesh given in Figure 7. In this problem we have three layers of prismatic elements. A non-conservative velocity field is defined at the nodes of the mesh, then projected conservatively onto the prisms. Local mass errors before and after postprocessing for each of the three layers are given in Figures 8, 9 and 10.

4 Conclusions and future work

There are several avenues for improvement of the present code. First, the code is still too slow for production work. A better linear solver needs to be developed, and the code needs to be ported to parallel platforms. Moreover, more complete testing on three-dimensional cases needs to be performed.

5 Acknowledgment

This work was supported in part by a grant of HPC time from the DoD HPC Modernization Program.

References

[1] T. Arbogast, C.N. Dawson and P.T. Keenan, *Efficient Mixed Methods for Groundwater Flow on Triangular or Tetrahedral Meshes*, Computational Methods in Water Resources X, (Peters et. al., editors), Kluwer (1994), pp. 3–10.

[2] F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, Springer-Verlag, New York, 1991.

[3] C.F. Cerco and T. Cole, “User's Guide to the CE-QUAL-ICM Three-Dimensional Eutrophication Model, Release Version 1.0,” Technical Report EL-95-15, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS, 1995.

[4] S. Chippada, C.N. Dawson, M.L. Martinez and M.F. Wheeler, *Projection Method for Constructing a Mass Conservative Velocity Field*, Comp. Meth. Appl. Mech. Eng. 157, pp. 1-10, 1998.

[5] P.T. Keenan, *TUF 2.5 User Manual*, TICAM Report 96-10, University of Texas at Austin, 1996.

[6] P.T. Keenan, *kScript User Manual*, Dept. of Computational and Applied Mathematics Tech. Report #95–02, Rice University, 1995.

[7] I.P. King and W.R. Norton, “Recent application of RMA's finite element models for two-dimensional hydrodynamics and water quality.” in *Finite Elements in Water Resources II*, C. A. Brebbia, W. G. Gray, and G. F. Pinder, eds., Pentech Press, London, 1978.

[8] R.A. Luettich, Jr., J.J. Westerink and N.W. Scheffner, “ADCIRC: An Advanced Three-Dimensional Circulation Model for Shelves, Coasts and Estuaries,” *Report 1, U.S. Army Corps of Engineers*, Washington, D.C. 20314-1000, December 1991.

[9] R.A. Raviart and J.M. Thomas, “A mixed finite element method for 2nd order elliptic problems,” *Mathematical Aspects of the Finite Element Method*, Lecture Notes in Mathematics, Springer-Verlag, New York, vol.606 (1977), pp.292-315.

[10] W. Tan, *Shallow Water Hydrodynamics*, Elsevier Oceanography Series, 55, Elsevier, Amsterdam, 1992.

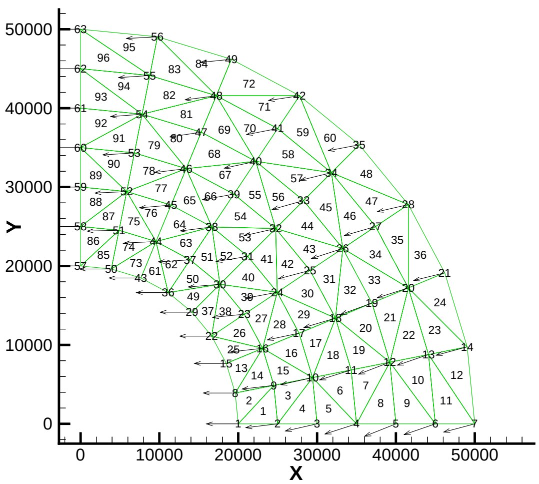

Figure 2: 2D test problem: the mesh and velocity field

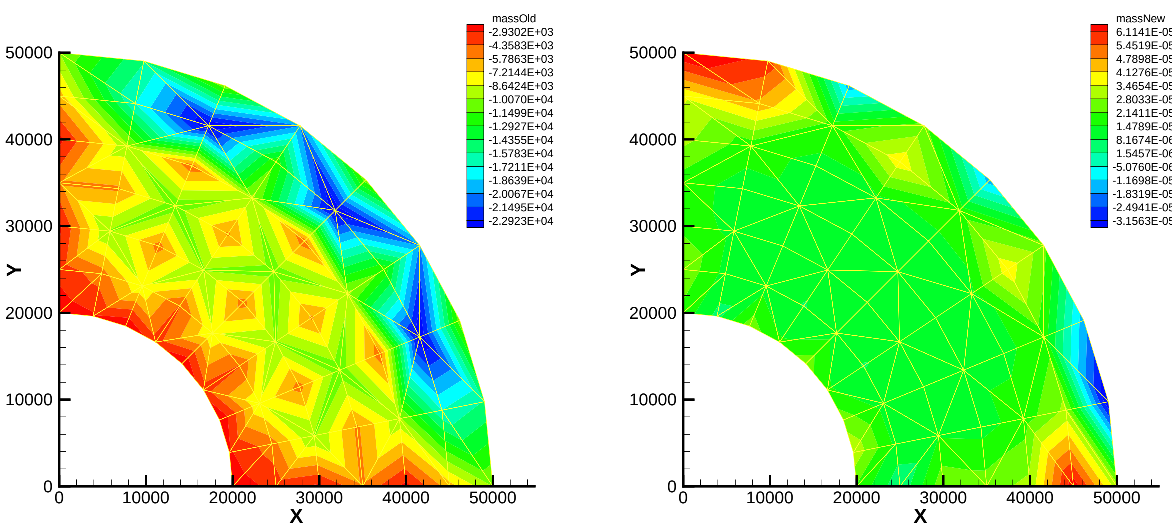

Figure 3: 2D test problem: local mass error. (Left) before postprocessing (Right) after postprocessing

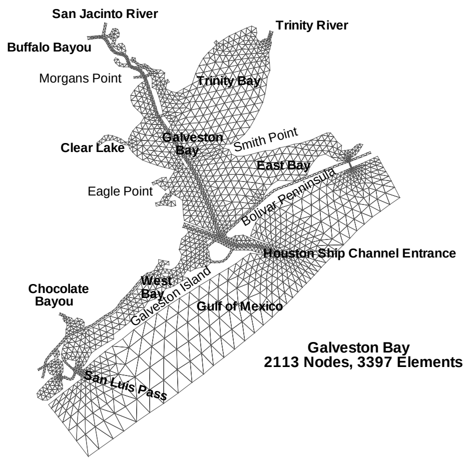

Figure 4: Galveston Bay and finite element mesh

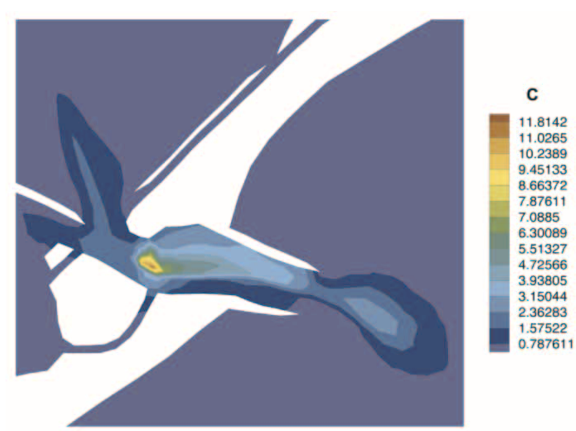

Figure 5: Contaminant concentration in area near spill

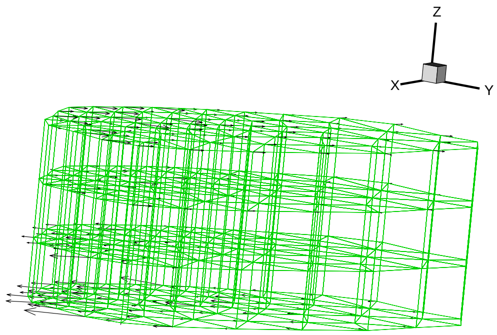

Figure 6: 3D test problem: the mesh and velocity field

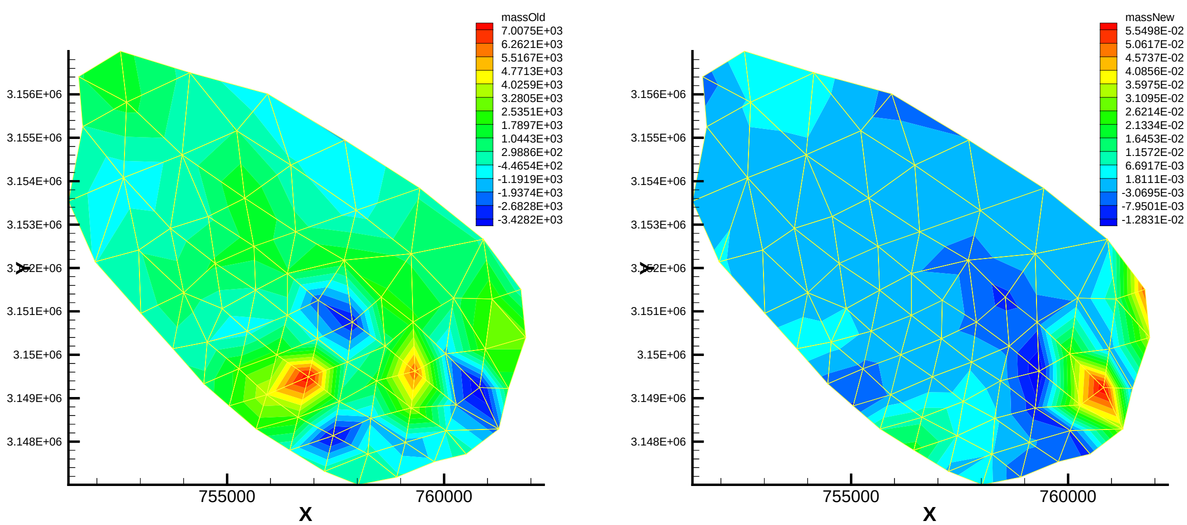

Figure 7: 3D test problem: local mass error on the bottom layer: (Left) before postprocessing (Right) after postprocessing

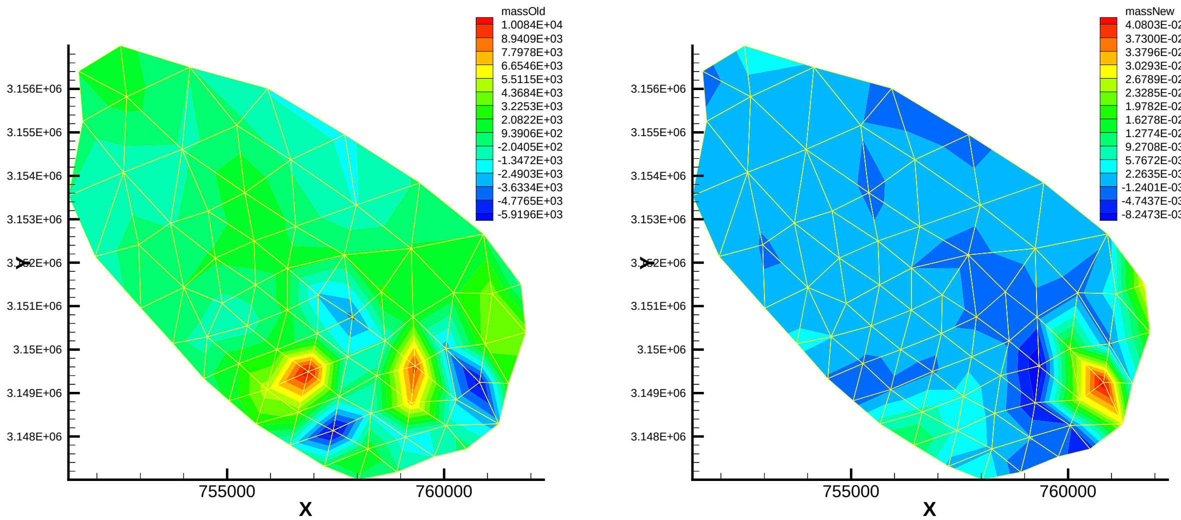

Figure 8: 3D test problem: local mass error on the 1st layer above the bottom: (Left) before postprocessing (Right) after postprocessing

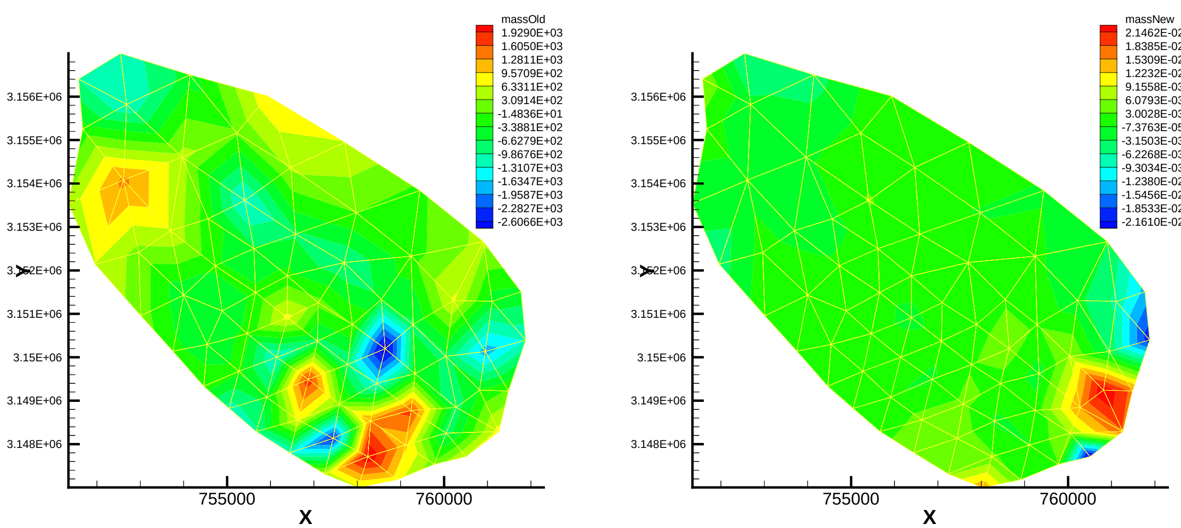

Figure 9: 3D test problem: local mass error on the surface layer: (Left) before postprocessing (Right) after postprocessing